Amendments to the Claims:

Status of Claims:

Claims 1-18 are pending for examination.

Claim 12 is canceled by the amendment.

Claims 1, 2, 3, 4, 6, 9, 11, 13, 16, 17, and 18 are amended herein.

Claims 1, 2, 6, 11, 13, 16, 17, and 18 are in independent form.

1. (Currently Amended) A system configured to interact with a virtual bus interface that is configured to produce a bus~~-type~~ transaction from a point-to-point ~~type~~ transaction, the system comprising:

a detection logic configured to detect whether the point-to-point transaction to be processed by the virtual bus interface includes a data~~-type~~ field that stores a data from which a value for a header~~-type~~ field in a bus~~-type~~ transaction can be produced; and

a decode logic operably connected to the detection logic, the decode logic being configured to extract the data from the data~~-type~~ field, to process the data into the value, and to selectively store the value in the header~~-type~~ field in the bus~~-type~~ transaction.

2. (Currently Amended) ~~The system of claim 1,~~

A system configured to interact with a virtual bus interface that is configured to produce a bus transaction from a point-to-point transaction, the system comprising:

a detection logic configured to detect whether the point-to-point transaction to be processed by the virtual bus interface includes a data field that stores a data from which a value for a header field in a bus transaction can be produced; and

a decode logic operably connected to the detection logic, the decode logic being configured to extract the data from the data field, to process the data into the value, and to selectively store the value in the header field in the bus transaction,

where the detection logic detects whether the point-to-point transaction includes a data~~-type~~ field that stores a data from which a value for a header~~-type~~ field can be produced by examining a transaction type associated with the point-to-point transaction.

3. (Currently Amended) The system of claim ~~1~~2, where the decode logic extracts the data from the data~~-type~~ field in a bit-field wise manner.

4. (Currently Amended) The system of claim 3, where upon detecting that a point-to-point transaction stores a data from which a value for a header~~-type~~ field can be produced, the detection logic generates a signal that is distributed to one or more of the decode logic and the virtual bus interface.

5. (Currently Amended) The system of claim 1, where the bus~~-type~~ transaction comprises a front-side bus transaction.

6. (Currently Amended) A virtual bus interface system, comprising:

a point-to-point transaction logic configured to receive a packet associated with a point-to-point transaction;

a bus~~-type~~ transaction logic operably connected to the point-to-point transaction logic, the bus~~-type~~ transaction logic being configured to produce a bus~~-type~~ transaction corresponding to the point-to-point transaction from the packet associated with the point-to-point transaction;

a detection logic operably connected to the point-to-point transaction logic, the detection logic being configured to detect whether the packet associated with the point-to-point transaction includes a data flit that encodes a non-memory-data value; and

a decode logic operably connected to the detection logic and the bus~~-type~~ transaction logic, the decode logic being configured to extract the non-memory-data value from the data flit, to decode the non-memory data value, and to selectively provide the decoded non-memory-data value to the bus~~-type~~ transaction logic.

7. (Original) The system of claim 6, where the detection logic detects whether the packet associated with point-to-point transaction includes a data flit that encodes a non-memory-data value by examining a transaction type associated with the point-to-point transaction.

8. (Original) The system of claim 7, where the decode logic extracts the non-memory-data value from the data flit in a bit-field wise manner.

9. (Currently Amended) The system of claim 6, where upon detecting ~~that~~ that the packet associated with the point-to-point transaction includes a data flit that encodes a non-memory-data value, the detection logic generates a signal that is distributed to one or more of, the point-to-point transaction logic, the bus~~-type~~ transaction logic, and the decode logic.

10. (Original) The system of claim 6, where the virtual bus interface produces a bus transaction for a front-side bus.

11. (Currently Amended) A computer configured with a virtual bus interface system, the virtual bus interface system comprising:

a point-to-point transaction logic configured to receive a packet associated with a point-to-point transaction;

a bus~~-type~~ transaction logic operably connected to the point-to-point transaction logic, the bus~~-type~~ transaction logic being configured to produce a bus~~-type~~ transaction corresponding to the point-to-point transaction from the packet associated with the point-to-point transaction;

a detection logic operably connected to the point-to-point transaction logic, the detection logic being configured to determine whether the packet associated with the point-to-point transaction includes a data flit that encodes a non-memory-data value; and

a decode logic operably connected to the detection logic and the bus~~-type~~ transaction logic, the decode logic being configured to extract the non-memory-data value from the data flit, to decode the non-memory data value, and to selectively provide the decoded non-memory-data value to the bus~~-type~~ transaction logic.

12. (Cancelled)

13. (Currently Amended) ~~The method of claim 12,~~

A method, comprising:

in a virtual bus interface, detecting a completion event associated with receiving a point-to-point transaction to be processed into a bus transaction by the virtual bus interface;

determining whether the point-to-point transaction includes a data flit that stores a value to be processed into a header field in the bus transaction, and

upon determining that a data flit stores a value to be processed into a header field in the bus transaction:

selectively extracting the value from the data flit; and

producing a header value from the extracted value,

where determining whether the point-to-point transaction includes a data flit that stores a value to be processed into a header ~~type~~ field includes examining a transaction type associated with the point-to-point transaction.

14. (Original) The method of claim 13, where the value is extracted in a bit-field wise manner from the data flit.

15. (Original) The method of claim 14, including:

establishing a decode function for a point-to-point transaction type in which a data flit encodes a non-memory data value; and

upon determining that a data flit encodes a non-memory data value, passing the data flit to an established decode function.

16. (Currently Amended) A computer-readable medium storing processor executable instructions operable to perform a method, the method comprising:

in a virtual bus interface, establishing a decode function for a point-to-point transaction type in which a data flit encodes a non-memory data value;

detecting a completion event associated with receiving a point-to-point transaction to be processed into a bus ~~type~~ transaction by the virtual bus interface;

determining whether the point-to-point transaction includes a data flit that stores a value to be processed into a header~~-type~~ field in the bus-type transaction by examining a transaction type associated with the point-to-point transaction; and

upon determining that a data flit stores a value to be processed into a header~~-type~~ field in the bus~~-type~~ transaction, passing the data flit to an established decode function, and storing a decoded value returned from the decode function.

17. (Currently Amended) A system, comprising:

means for determining whether a point-to-point transaction available to a virtual bus interface includes a data flit that stores non-memory-data information that will be stored in a bus~~-type~~ header~~-type~~ field;

means for bitwise field extracting the non-memory-data information from the data flit; and

means for decoding the extracted non-memory-data information and making the decoded non-memory-data information available to a virtual bus interface logic configured to produce a header~~-type~~ field for a bus~~-type~~ transaction.

18. (Currently Amended) A set of application programming interfaces embodied on a computer-readable medium for execution by a computer component in conjunction with producing a header~~-type~~ field for a bus~~-type~~ transaction from non-memory-data information stored in a data flit in a point-to-point type transaction, comprising:

a first interface for communicating the data flit that encodes the non-memory-data information;

a second interface for communicating a non-memory-data value extracted from the data flit; and

a third interface for communicating a header~~-type~~ data value for the bus~~-type~~ transaction, where the header~~-type~~ data value is produced from the non-memory-data value extracted from the data flit.